ARL

US Army Research Laboratory

A Heuristic Evaluation of the Generalized Intelligent Framework for Tutoring (GIFT) Authoring Tools

by Scott J Ososky and Robert A Sottilare

NOTICES

Disclaimers

US Army Research Laboratory

A Heuristic Evaluation of the Generalized Intelligent Framework for Tutoring (GIFT) Authoring Tools

by Scott J Ososky
Oak Ridge Associated Universities, Oak Ridge, TN

Robert A Sottilare
Human Research and Engineering Directorate (HRED)

REPORT DOCUMENTATION PAGE			*Form Approved* *OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**			

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)	
March 2016	Special Report	July 2015–December 2015	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A Heuristic Evaluation of the Generalized Intelligent Framework for Tutoring (GIFT) Authoring Tools	
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
Scott J Ososky and Robert A Sottilare	
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
US Army Research Laboratory ATTN: RDRL-HRT-T Aberdeen Proving Ground, MD 21005-5425	ARL-SR-0353

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The purpose of this report is to present the findings of a usability evaluation of the course authoring tools found within the Generalized Intelligent Framework for Tutoring (GIFT, version 2014–3×). Specifically, a heuristic evaluation was conducted on the GIFT authoring tool interfaces, as well as the process by which the software is acquired and installed. This report describes the specific findings of the evaluation, along with recommendations for interface improvements. Finally, the findings are discussed as recommendations that can be applied throughout the GIFT authoring tools. Those recommendations for future consideration and improvement include matching the design of the system and interface language to users' mental models, assuring internal consistency of user interfaces and object labels, assisting users with content organization and error prevention, and providing a current and consistent help and documentation experience throughout the authoring tools.

15. SUBJECT TERMS
GIFT, course authoring tools, heuristic evaluation, usability, interface, interaction, user experience

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Robert A Sottilare
a. REPORT	b. ABSTRACT	c. THIS PAGE	UU	50	19b. TELEPHONE NUMBER (Include area code)
Unclassified	Unclassified	Unclassified			407-208-3007

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

1. Introduction

The purpose of this report is to document and report the findings of a heuristic evaluation of the Generalized Intelligent Framework for Tutoring (GIFT) authoring tools. The GIFT authoring tools consist of a series of software interfaces that support users in the creation of adaptive tutors (also known as intelligent tutoring systems [ITSs]) built upon the principles of the framework. The purpose of the heuristic evaluation was to identify and document potential usability concerns that may negatively impact users' experiences when interacting with the GIFT authoring tools. A primary goal of GIFT is to make adaptive tutoring available and usable by the masses. Expected outcomes from this evaluation include better user experiences through easier organization of domain knowledge and faster, more efficient authoring.

1.1 Overview of GIFT, Authoring Tools

The GIFT concept document (Sottilare et al. 2012) describes GIFT as "an empirically-based, service-oriented framework of tools, methods, and standards to make it easier to author computer-based tutoring systems (CBTS), manage instruction and assess the effect of CBTS, components and methodologies". Simultaneously, GIFT is an open-source research project and public-facing application. GIFT is currently under development and includes a number of technologies, features, tools, and methods intended to support a variety of users including instructional designers, authors, instructors, researchers, and learners. For instance, the framework incorporates the following: models of domain knowledge (e.g., content, feedback, and remediation); pedagogical methods based on best instructional practices from the literature; interfaces to support sensor integration; gateways to support interoperability with external applications; and models of learner states and traits. As such it is important to recognize that GIFT is not simply a set of authoring tools. However, the evaluation of the GIFT authoring tools is the focus of this report.

Previously, GIFT courses were created exclusively by writing and/or editing schema within a file formatted as eXtensible Markup Language (XML). At the time of this evaluation, the stable and experimental versions of GIFT (2014-2 and 2014-3X, respectively) provided browser-based interfaces to facilitate semi-automated user creation of the XML output. These new interfaces are intended to support the following overall goals for GIFT (Sottilare et al. 2013): decrease the effort and skill threshold required for authoring ITSs; support users in organizing their domain content and knowledge; support effective instructional strategies;

allow for rapid prototyping of ITSs; leverage standards for integration of external content (media and software applications); and promote content reuse and interoperability through standards. These goals were considered in the heuristic analysis of the authoring experience for GIFT users.

1.2 Overview of Heuristic Evaluation

Jakob Nielsen's (1994, 1995) 10 general principles for user interface (UI) design served as the guiding principles or heuristics for this evaluation. Nielsen's heuristics have been applied to the context of the GIFT authoring tools and are presented in the following list for reference. Potential design issues for the GIFT authoring tools are described in Section 2 of this report.

- **Visibility of System Status:** The system should always keep users informed about what is going on within the system through appropriate feedback within reasonable time to affect decisions by the user. In the GIFT authoring process, the user should understand where they are in terms of progress in authoring a tutor.

- **Match between System and the Real World:** The system should speak the users' language, with words, phrases, and concepts familiar to the user, rather than system-oriented terms or jargon. The system should follow real-world conventions and make information appear in a natural and logical format. In the GIFT authoring process, there should be standard, consistent terms (a glossary) to describe elements in the authoring process and their relationship to each other (i.e., an ontology).

- **User Control and Freedom:** Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. The GIFT authoring tools should support Undo and Redo functions.

- **Consistency and Standards:** Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions. Ideally, GIFT authors should be able to develop a GIFT tutor within their native platform (e.g., virtual training simulation) to support goals of consistency and commonality with the "real world".

- **Error Prevention:** Even better than good error messages is a careful design that prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action. Research is

needed to identify common errors that result from using the GIFT authoring tools during the ITS authoring process. This will allow GIFT to monitor for known errors or omissions.

- **Recognition Rather Than Recall:** A goal for user-centered design is to minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of a dialogue or process to another. Instructions for use of the system should be visible or easily retrievable whenever needed. The GIFT authoring tools UI should provide a dashboard with essential information to support the authoring process.

- **Flexibility and Efficiency of Use:** Accelerators, usually unseen by the novice user, may often speed up the interaction for the expert user such that the system can cater to a range of users from inexperienced novices to moderately experienced journeymen to highly experienced experts. The system should adapt the amount and type of information available to the user based on their capabilities, needs, and limitations, and change information availability as the user's expertise in using the authoring tools grows.

- **Aesthetic and Minimalist Design:** Dialogues should contain only germane information and not contain information that is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. Germane information should only be visible to the user when they need it and should always be available to the user when they need it.

- **Help Users Recognize, Diagnose, and Recover from Errors:** Error messages should be expressed in plain language (e.g., not just error codes) that precisely indicate the problem, and constructively recommend a solution. All GIFT-based tutors should be validated. Thus, if information that is critical to the function of the ITS is missing when the author believes the authoring process is completed, the GIFT authoring tools should alert them and take them to the next step in the process.

- **Help and Documentation:** Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be unnecessarily long. An essential part of the help process for authoring ITSs will be the availability of exemplar tutors that allow the author to understand how GIFT works by providing a concrete example.

1.3 Evaluation Methodology

The initial heuristic evaluation of the GIFT authoring tools was completed in 2015, as part of the overall development plan for the GIFT authoring experience. Based on the relevant documentation and discussion with GIFT stakeholders, the users of GIFT authoring tools could potentially include instructional developers, instructors, subject matter experts, and researchers. For purposes of this evaluation, the GIFT authoring tools were evaluated from the instructor point-of-view (including teachers and facilitators). We anticipate that additional heuristic evaluations will be needed as changes based on authoring stakeholders are implemented within GIFT.

The instructor user group was conceptualized to possess the following characteristics: low subject matter expertise (i.e., not domain experts), moderate understanding of instructional design principles, low to moderate computer knowledge and comfort with technology. It is anticipated that instructors will use GIFT with the goal of creating or managing computer-based adaptive tutoring content. It is anticipated that instructors will typically leverage existing course materials (e.g., text, assessments) and training application content (e.g., slide presentations, 3-dimensional simulation software). The completed adaptive tutoring application would be exported and shared with other GIFT users for training (i.e., students) or evaluation and modification by other authors.

This evaluation of GIFT authoring tools includes the process of acquiring and installing GIFT software, as well as authoring and exporting a course generated with the GIFT authoring tools. The version of GIFT evaluated in this report was version 2014-3X. Although this version is designated as an experimental version, the stability of the authoring tools was deemed sufficient for the current evaluation.

2. Heuristic Evaluation Findings

This section of the report outlines the tasks required to support the authoring of an ITS using GIFT. It includes the task, known issues, and recommended solutions to improve the usability of the GIFT authoring tools.

2.1 Task: Acquiring, Installing, and Launching GIFT Software

2.1.1 Issue: Software Is Difficult to Locate and Download from the GIFT Website

GIFT software is acquired from the website www.gifttutoring.org. The page does not contain a clear download link (Fig. 1). In order to acquire the software, users must first log in (or create a new account), then click on the Files tab. The links on the file page are identified by file name, and it is not clear which file or series of files are needed to acquire the correct GIFT install package (Fig. 2).
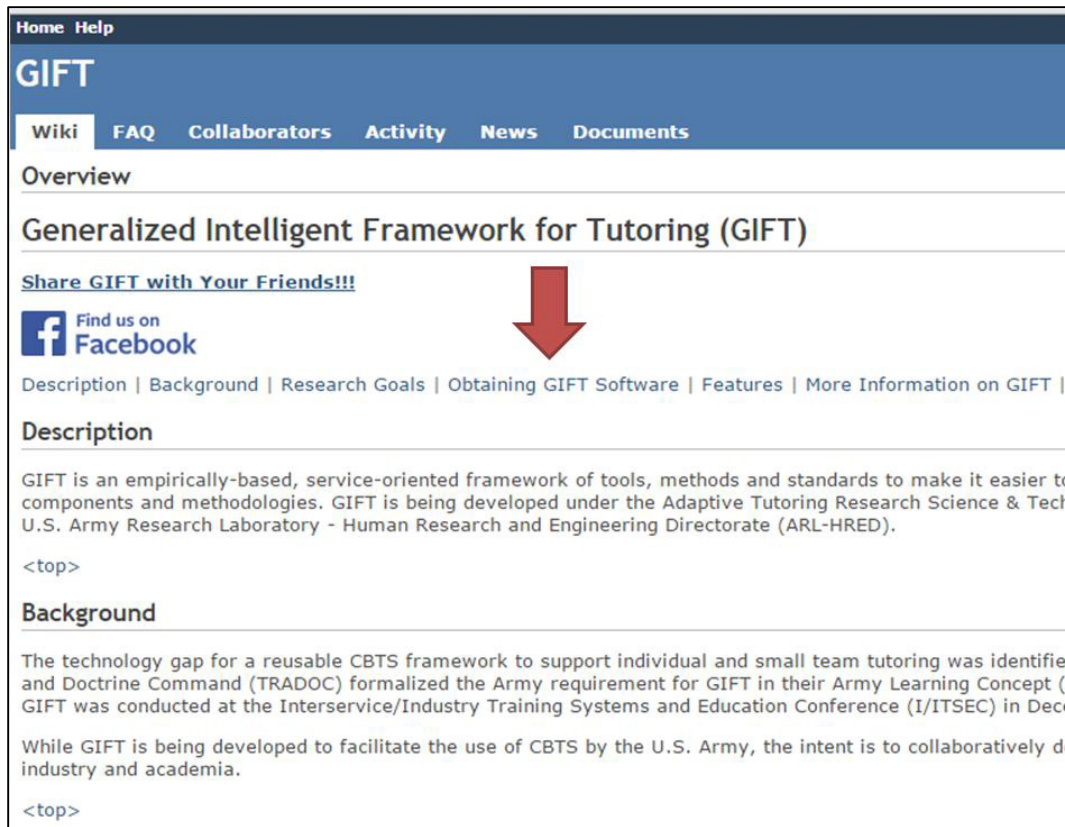
Fig. 1   GIFT website with link to acquire software notated

Fig. 2   File list on gifttutoring.org with current release notated

This issue is not unique to the authoring tools. However, because of the relatively complex process by which GIFT software is acquired and installed, users may become confused before they can run the software.

Heuristics: visibility of system status; consistency and standards

Recommendations:

- The link to download software should be clearly identified on the home page of www.gifttutoring.org. For example, a download icon can accompany a text link in the main navigation bar on the page.

- "Download" may be a more common/recognizable link compared to "files" or "Obtaining GIFT Software". [Note: A "download" button was determined to be appropriate and has since been implemented on the GIFT website. This button functions to alert the user to log in (if this has not been done), then initiates the download process. An option to allow the user to log in and use GIFT online (i.e., no download required) is also desirable and has since been implemented.]

- Downloads on the file page could be labeled with descriptive text to indicate main install files, extensions, applications, and other pertinent information.

2.1.2 Issue: Unclear Process for Starting GIFT Software Installation

Users may not understand how to install the software once it is acquired from the website. Users are typically familiar with self-installing applications (e.g., exe files) for desktop applications, and completely automated install processes for mobile applications (e.g., digital application-store downloads). By comparison, the GIFT installation requires users to unzip a file to a directory of their choosing and then must run the "installGIFT.bat" file in order to begin the installation.

Heuristics: help and documentation; error prevention

Recommendations:

- Add features to the zip file that allow it to specify a target folder name, but allow the user the option to change the target folder and its placement on the local drive.

- Provide a single install file (e.g., exe) for users that have sufficient permissions to install software on their computers; this would also be consistent with single install files for GIFT exported courses and experiments.

2.1.3 Issue: Limited Documentation for Installing GIFT Using the Zip File

Users may not be able to locate the help and documentation for installing GIFT using the zip file. Only 2 files appear in the root folder of the zip file: "installGIFT.bat" and "GIFT_2014-3x_Release_Notes.pdf". The batch file will not run correctly directly from the zip file. The release notes file has a Quick Start guide; however, there are only 2 steps listed, and the second step refers the user to another file in the folder structure. The Quick Start guide also appears on page 5 of the document. The installation document suggests that only Windows 7 is supported. Windows 7 is shortened to "Win7". The documentation supporting GIFT installation may be confusing to users with limited computer knowledge.

Heuristics: help and documentation; consistency and standards

Recommendations:

- Win7 appears to be an unofficial term. Win7 should be spelled out as Windows 7 in documentation.

- Provide support information within documentation with respect to all operating systems that the GIFT software is compatible with.

- Articulate all relevant installation instructions within the release notes file.

- Add a plain text file to the root of the zip folder such as "README_FIRST.txt" and repeat the full list of installation instructions in this file.

2.1.4 Issue: Inconsistent Branding and Version Labeling in GIFT

A number of different GIFT logos appear between the GIFT website, software application, and tutor-user interfaces (TUIs) (Fig. 3). Users may be confused by the various logos and may mistake the appearance of different logos as mismatched software versions, or users may believe that they were diverted to a phishing website.

Fig. 3   GIFT logos encountered during interaction

Additionally, a different version number appears in the installer when running the batch file for GIFT-2014-3X. The version number in the install wizard labels the software as "GIFT 5.1" (Fig. 4). Users may believe that they somehow acquired an incorrect software version.
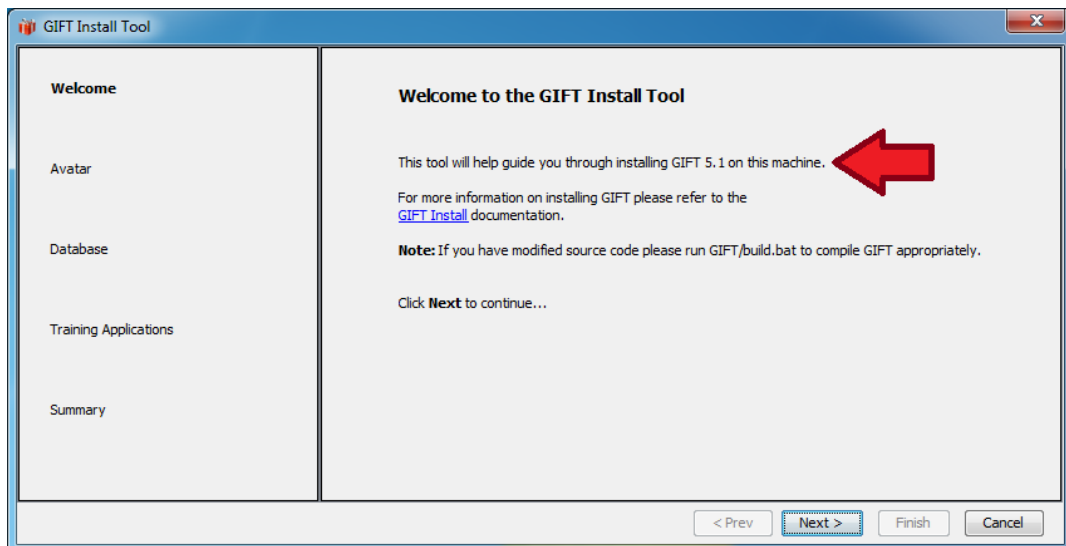
Fig. 4   GIFT version inconsistent with zip file name

Heuristic: consistency and standards

Recommendation:

- Use a single GIFT logo for the website, software, and TUI. Variations on the same base logo could be used to indicate subproducts of GIFT, where appropriate (as of the writing of this report, the US Army Research Laboratory [ARL] has adapted a set of approved GIFT logos); version names and numbers should be consistent throughout the user experience.

2.1.5 Issue: Awkward Installation Procedure

There are 2 separate steps in the GIFT install procedure that may require the user to terminate and restart the installation process to acquire and install third-party software. These steps are labeled as "Avatar" and "Training Applications". The "Database" stage of the installation process only displays an information line of text and appears to be unnecessary (assuming no errors). The configuration of the third-party avatar appears to be the only step of the process that is on the critical path. Finally, the "Training Application" stage only lists 2 third-party extensions, which may lead the user to believe these are the only 2 applications compatible with GIFT.

Heuristics: recognition rather than recall; help and documentation; help users recognize, diagnose and recover from errors

Recommendations:

- Provide a mechanism to automatically download and install the media semantics character (avatar), should the user choose to use it.

- Provide more descriptive text for the steps of installation, such as "Configure Avatar" or "Build Database".

- Steps that are not on the critical installation path can be removed and relocated under a set of options within the main program, postsetup.

- Inform users that it is possible to install the avatar/training applications at a later time if not done during installation, provide instructions for this scenario.

2.1.6 Issue: Confusing Message at the End of Installation

After the user progresses through each of the stages in the GIFT Install Tool, a pop-up appears that displays installation progress. Once this process has completed the message box will indicate that the installation has completed; however, the only option for the user to click on is Cancel (Fig. 5). Instead, it is

not even possible to dismiss this box until the Readme text file has been closed. This may be confusing to users, who may think that clicking Cancel will undo the installation. Users might also want to allow the Readme file to remain open, while closing the installation dialog windows.
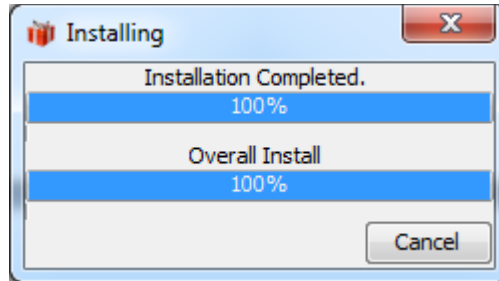
Fig. 5   GIFT Install Tool progress pop-up

Heuristics: visibility of system status; consistency and standards

Recommendations:

- Update the pop-up message box to include an "Ok" button, or allow the progress bar to close gracefully.

- Allow the dialog box to be dismissed independently of the readme text pop-up.

2.1.7   Issue: No Obvious Way to Launch the GIFT Authoring Tools after Installation

Once the GIFT software has been installed, it is not obvious where the authoring tools are located. The installation process does not place any icons on the desktop or Start menu. The additional shortcut placed in the GIFT installation folder, "launchGIFT.bat" leads the user to the TUI login screen, not the authoring tools. The TUI does not present information to the user that would assist them in locating the authoring tools.

The authoring tools are currently accessed by navigating the file system in the following path "[GIFT install folder]\GIFT\scripts\launchControlPanel.bat". This script launches a Control Panel window, in which the second tab in the window is labeled "Authoring Tools".

Additionally, the Readme file that appears after GIFT is launched does not display the full path name to reach the Control Panel. The Readme file states, "Doing more advanced things, like DEVELOPING YOUR OWN MATERIAL [*sic*]… can be accessed via the 'launchControlPanel.bat' script in the scripts

directory." For the users that do examine the Readme file, the Control Panel (and authoring tools) may still be difficult to locate for new users.

Heuristics: help and documentation; recognition rather than recall

Recommendations:

- Create a shortcut on the desktop or in the install folder leading to the Control Panel.

- Consider adding a separate shortcut that provides the same functionality as the Control Panel Batch file but is labeled "launchAuthoringTools.bat". This file would open the same Control Panel but would default to the Authoring Tools tab.

- Articulate the full path to the appropriate batch file in the readme file.

2.1.8 Issue: Unintuitive/Unfamiliar Layout of Authoring Tool Control Panel

The GIFT Control Panel provides a series of buttons under Authoring Tools (Fig. 6). Clicking each of these buttons launches a separate editing/authoring interface. Some of these interfaces are different from the others (e.g., some links launch XML editors, others launch web-based interfaces). The Control Panel interface lacks descriptive text aside from the button labels. No Help function is available on the Control Panel Menu. The names of the editing tools may be unfamiliar to inexperienced users of GIFT. The button names have inconsistent labels: 2 of the buttons are not labeled as tools (i.e., workbench, system); one of the buttons has an acronym in the name instead of spelled out completely. The similarly named Gift [sic] Authoring Tool and Course Authoring Tool launch different editors for the same functionality (i.e., course transitions). Users may be confused by the various tools in the interface, unfamiliar acronyms, or order in which tools should be accessed.
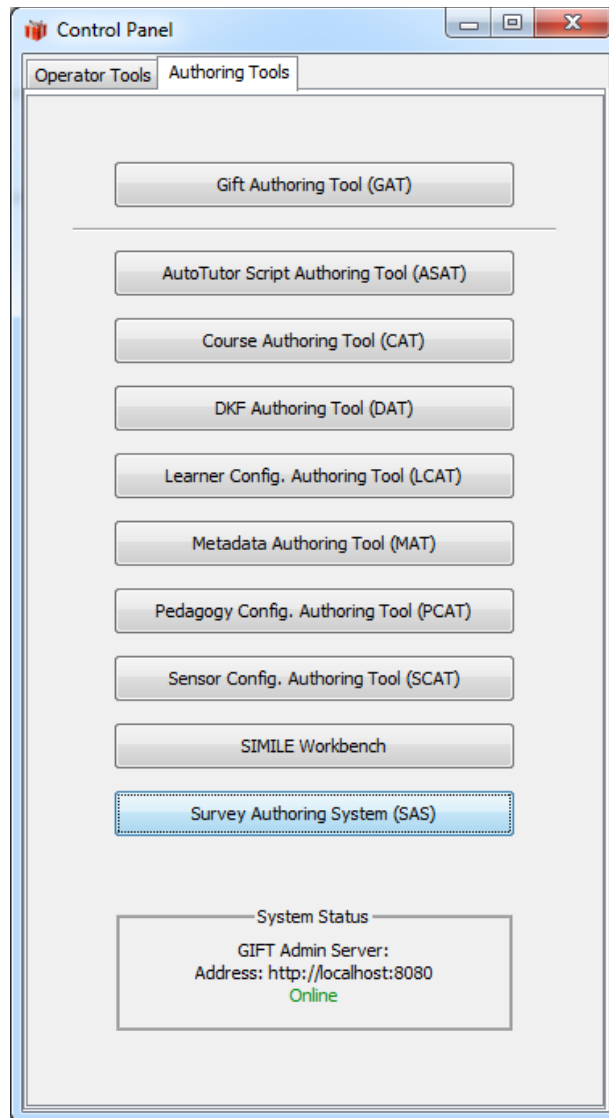
Fig. 6   **The authoring tools tab of the GIFT control panel**

Heuristics: match between system and the real world; consistency and standards; recognition rather than recall; help and documentation

Recommendations:

- Add a menu option that provides quick access to help documentation.

- Add dialogue to each button that appears on button mouse-over (hover).

- Descriptive text could indicate the type of program that will launch upon selection (web browser, XML editor tool, proprietary software).

- Consider standardize the naming convention of the authoring tools, systems, and workbenches.

- Consider removing the acronyms from the buttons if they are not used elsewhere.

2.1.9 Issue: Inconsistent Reference to GIFT in Control Panel

An inconsistent reference to GIFT appears in the Authoring Tools tab of the Control Panel. The top button refers to the "Gift [*sic*] Authoring Tool" (Fig. 6). This is inconsistent with the standard spelling of GIFT, the acronym, in all caps.

Heuristic: consistency and standards

Recommendation:

- The term GIFT should consistently appear in all caps throughout the experience.

2.2 Task: Creating a Course Using the GIFT Authoring Tools via the GIFT Dashboard

2.2.1 Issue: Name Mismatch between Gift Authoring Tool and Browser UI

Users may not understand the relationship between the Gift Authoring Tool and the GIFT Dashboard (Fig. 7). The Control Panel provides a link to the Gift Authoring Tool (GAT); however, the page that appears has the title, "GIFT Dashboard". No references to "authoring tools" appear on this initial web-interface. Users may also be confused by the phrase Dashboard, which might be more closely associated with a student's or instructor's course-record interface.
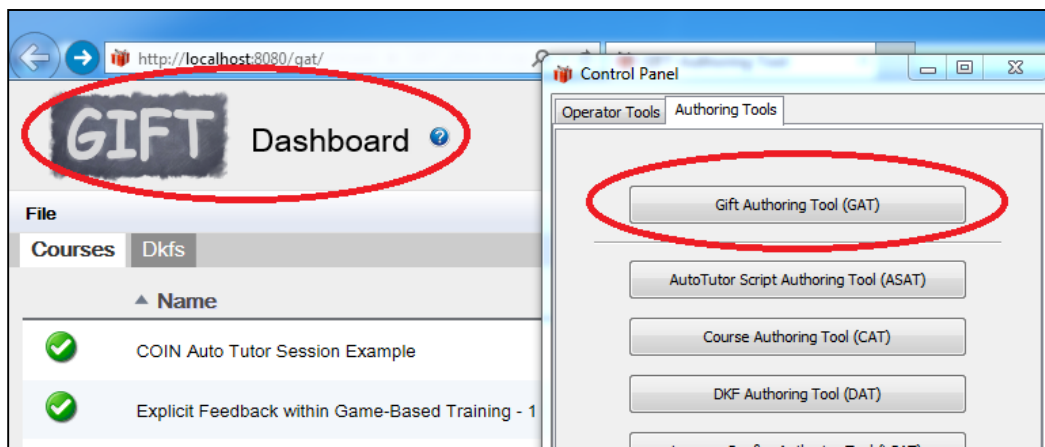
Fig. 7   GIFT dashboard and control panel labeling

Heuristic: consistency and standards

Recommendation:

- Rename the GIFT Dashboard as the GIFT Authoring Tool; names of tools, pages, etc., should appear consistent throughout the interfaces.

2.2.2 Issue: No Obvious Navigation through the Authoring Process from the Dashboard

Users may be unfamiliar with the workflow for authoring a new course from the GIFT Dashboard. The website does not provide an immediately recognizable user path for creating a new course. To create a new course, users must click on File and select New Course. If the user is on the "Dkfs" [*sic*] tab, the options in the File menu change, although "File" appears above tabs in the visual hierarchy (UI) of the page.

Heuristics: consistency and standards, recognition rather than recall, match between system and real world

Recommendations:

- Provide a link to allow the user to create a New Course from the home page.

- Move the "File" UI element below the tabs (Courses, domain knowledge files [DKFs]) on the Dashboard.

2.2.3 Issue: Awkward Access to Documentation from GIFT Dashboard

The web-based GIFT authoring tools provide access to help and documentation information via a question mark icon at the top of each page near the page title (Fig. 8). Clicking this icon will result in the documentation attempting to open with the same browser tab. This may cause the user to lose progress made in course authoring. Additionally, an error window explains that navigating away from the page may result in a loss of data; however, the message text does not correctly reference the selectable options on the message box. This issue may also occur when clicking the GIFT image at the top-left hand corner of each page.
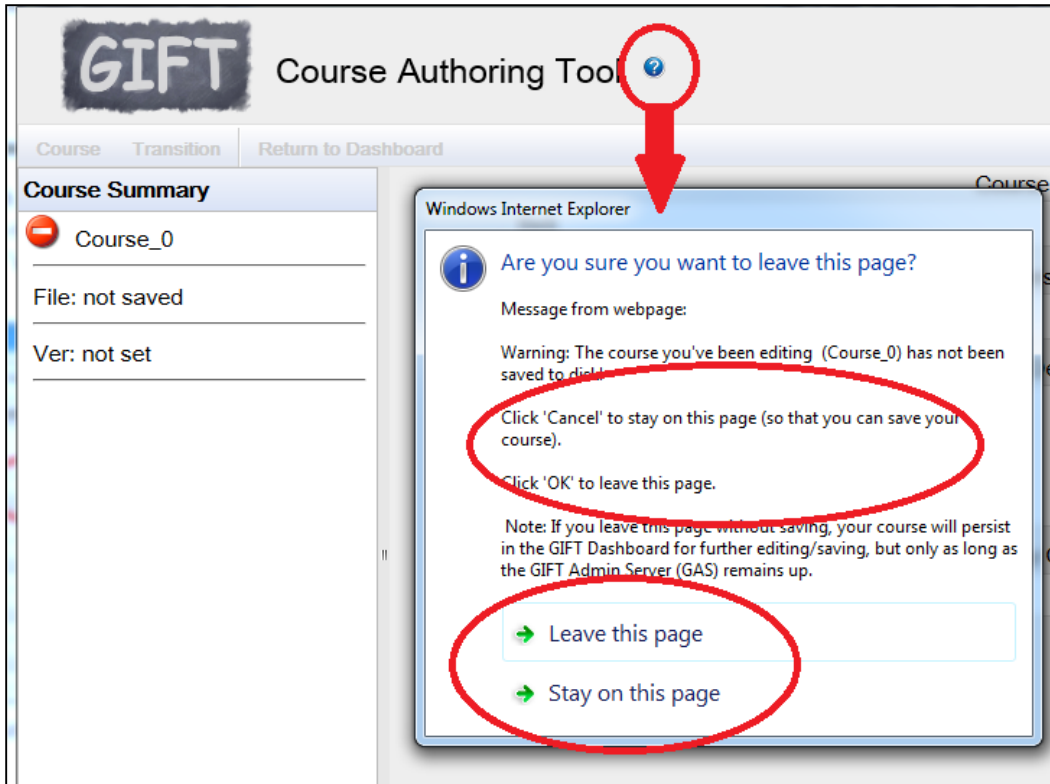
Fig. 8   Access to documentation results in pop-up message

Heuristics: error prevention; help and documentation

Recommendations:

- Clicking the links at the top of the page opens in different tabs or windows.

- Consider creating a Windows-style pop-up Help menu (e.g., F1 shortcut menu).

- Warning message instructions matches available user options in pop-up.

2.2.4   Issue: Difficult to Navigate in GIFT Help and Documentation

Users may experience difficulty in effectively using web-based GIFT documentation. The current documentation contains a number of pages, some of which reference information contained within other pages via hyperlink (Fig. 9). Accessing help is not contextual (e.g., clicking Help within the DKF Authoring Tool (DAT) does not go directly to documentation related to the DKF authoring process), and the documentation does not have a search feature. This places the burden on the user to navigate to the correct page and execute a manual keyword search. Topics are generally grouped together in a logical fashion, but there may

be cases in which information on a specific topic appears within multiple pages. For instance, the bulk of survey authoring information appears within the Survey Authoring System (SAS) Instructions page; however, information regarding migration of survey elements to new versions of GIFT appears within the Authoring Guide page.
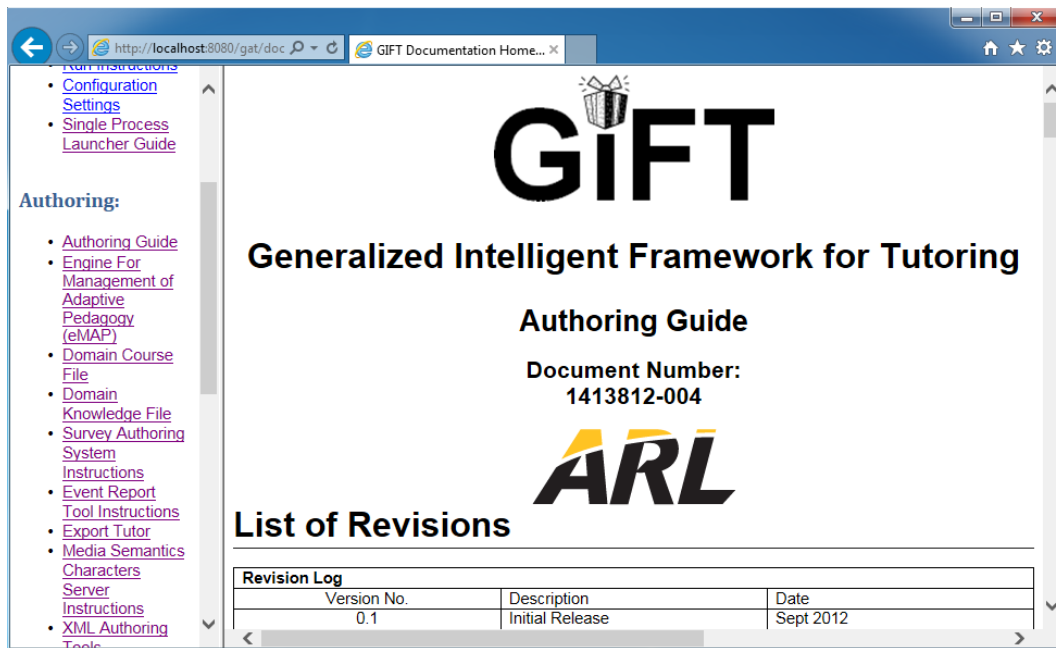
Fig. 9   Web-based GIFT help and documentation

Heuristic: help and documentation

Recommendations:

- Add a built-in search function to the documentation.

- Accessing help from a particular authoring tool should go directly to documentation on that specific authoring tool and/or concept.

- Consider hosting a live, online version of documentation within a wiki-style interface that would support search in addition to crowd-sourced updates and version control.

2.2.5   Issue: Documentation Not Specific to Web-Based Course and DKF Authoring Tools

Users may encounter difficulty in using the documentation as an aid for Course and DKF authoring with the web-based UI. The documentation is not specific to the web interface. By comparison, the SAS has documentation within the Help file that is specific to the web-based UI.

Heuristic: help and documentation

Recommendation:

- Provide specific documentation for all web-based tools and interfaces.

2.2.6 Issue: Unfamiliar File Management Workflow within the Course Authoring Tool

Users may be confused by the process to save courses within the Course Authoring Tool. Currently, users must click Commit at the bottom right-hand corner of the interface, before the options to save are enabled in the top left-hand corner of the page. Once changes are made within the Course Authoring Tool, the main menu bar will remain inactive until the Reset or Commit button is clicked. There is no visual indication explaining why the menu bar is inactive. Users may also mistake the Commit action for file saving, though the 2 functions are different.

Heuristic: visibility of system status

Recommendations:

- Do not deactivate the menu bar, add "loss of data" warning messages where needed.

- Clicking Save (or Save As) within the menu bar will automatically Commit all uncommitted changes.

- Clicking New Course, Return to Dashboard, etc. will prompt the user with a warning of a destructive action.

- A status indicator appears within the UI when new changes have not yet been committed.

2.2.7 Issue: No Folder Creation When Saving within the Course Authoring Tool

User may encounter difficulty with file management saving within the Course Authoring Tool. When using the Save As function, a window appears that is different from the standard Windows UI file browser (Fig. 10). From this dialog, it is only possible to name the course file and save it. By default this will place the course file within the "Domain" folder in the GIFT file system. However, GIFT documentation recommends placing course elements within a subfolder within the

Domain folder. This operation is not possible with the current UI. Later, if a user exports the course, all files in the domain folder (whether relevant or extraneous) will also go into the exported course package along with the targeted subfolder.
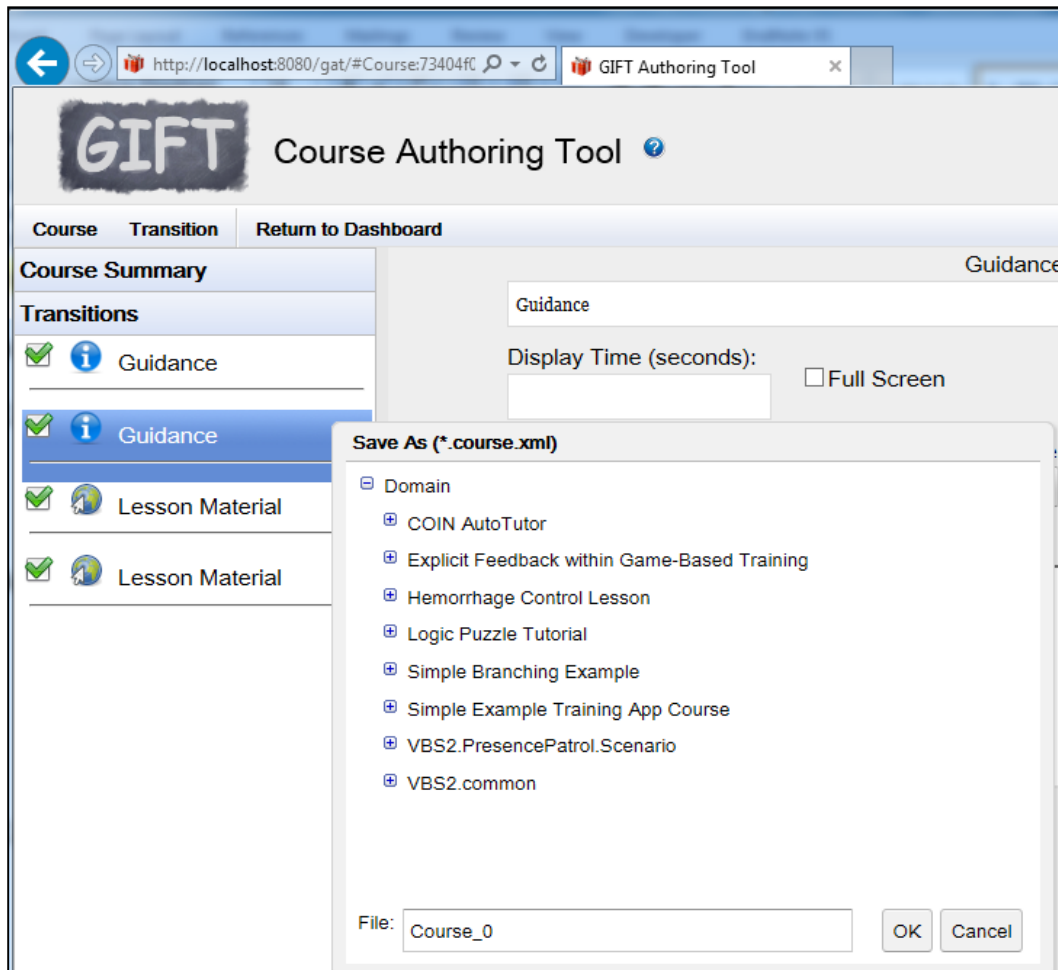
Fig. 10 Save as dialog in course authoring tool

Heuristics: visibility of system status; error prevention; flexibility and efficiency of use

Recommendations:

Three potential solutions are as follows:

- Use a standard Windows file explorer for saving courses.

- Allow users to create new folders within the current UI.

- Create "project" folders by default (automatically) to organize course materials.

2.2.8   Issue: Awkward Interface to Select Lesson Material Resources

Users may encounter difficulty when using the Lesson Material transition to insert a media element into a course. The user is required to specify the location of a media item by entering a Media Uniform Resource Identifier (URI) into the corresponding text area (Fig. 11). Users may not be familiar with what a URI is. (The Guidance transition, by comparison provides a similar UI with more supportive user functions.) There is also considerable room for error as the user is responsible for manual entry of the URI into the text field. The mouse-over help text does not provide new information to the user.
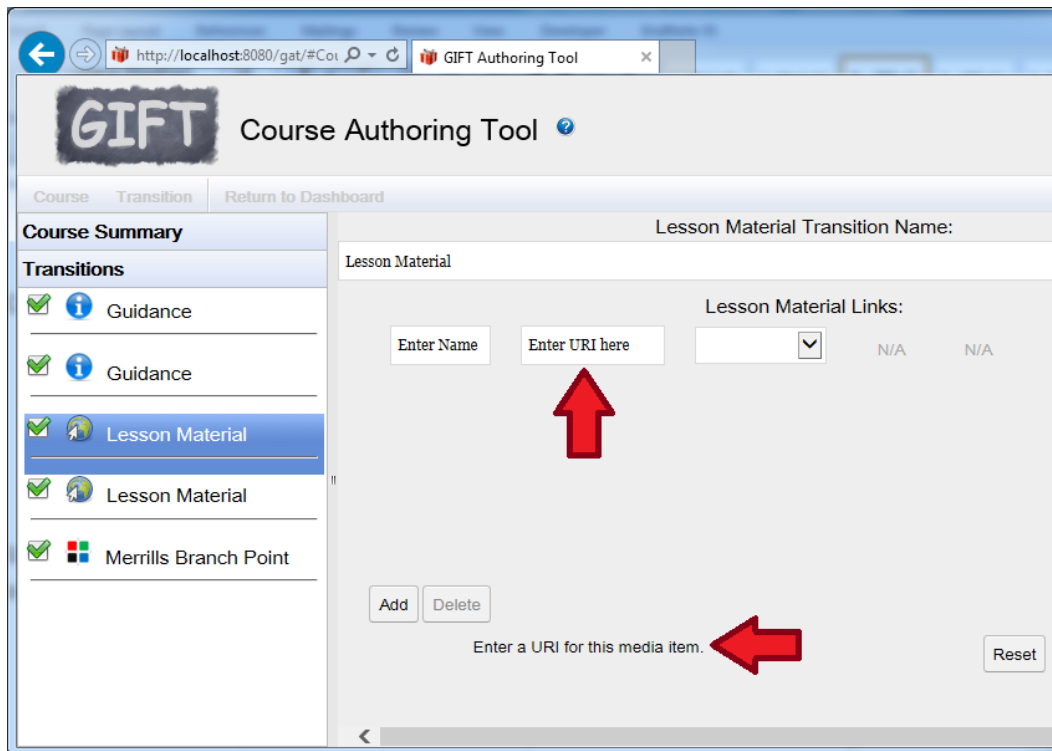
Fig. 11   Lesson material transition with help text

Heuristics: error prevention; recognition rather than recall; match between system and real world

Recommendations:

- Add option to specify either local file or web-based resources.

- Add a browse button to help users select the correct resources.

- Provide additional information in help text to indicate the types of URI supported and/or provide examples (e.g., "www.google.com" or "c:\documents\lesson.pdf").

- Provide simple error checking for proper URI formats (e.g., "http::/www.google.com is not a valid URI format").

2.2.9 Issue: No Preview Option Available in the Course Authoring Tool

Users may become frustrated when creating courses within the Course Authoring Tool without the ability to preview their course edits. Users may want to evaluate the look and functionality of a single transition, or inspect the entire course flow. To preview courses currently, users are required to launch the "student" version of GIFT, log in, and run their desired course.

Heuristics: visibility of system status, user control, and freedom

Recommendation:

- Allow users to preview single transitions or an entire course from within the Course Authoring Tool.

2.2.10 Issue: Limited Support for Undo/Redo in the Course Authoring Tool

Users may not understand how GIFT supports the undo functionality. Changes within a transition can be undone by clicking the Reset button at the bottom of the frame. (However, there is no positive confirmation of a potentially destructive action sought from the user when clicking Reset.) Changes to a course can be undone by going into the course menu and selecting "discard changes". Users may be more familiar with the UI convention of going to an Edit menu and selecting Undo. There is currently not an option in GIFT to redo an edit/addition.

Heuristics: user control and freedom, error prevention

Recommendations:

- Standardize the toolbar with familiar menu options: File, Edit, Add, Insert, and/or Help. Add an Undo function to the Edit menu.

- Provide a warning message and ask for positive confirmation when clicking the Reset button.

2.2.11 Issue: Confusing System Error Messaging Throughout GIFT Authoring Tools

Users may become frustrated with the types of error messages that can potentially be encountered while using GIFT authoring tools. Error messages throughout the tools and applications are inconsistent in their presentation: some error messages are relatively user-friendly, written in plain language with steps the user can take

to resolve the issue; other error messages are technical in nature, containing system-level error codes. Error messages in GIFT do not always provide a clear path to resolving the issue described as shown in the examples in Fig. 12.
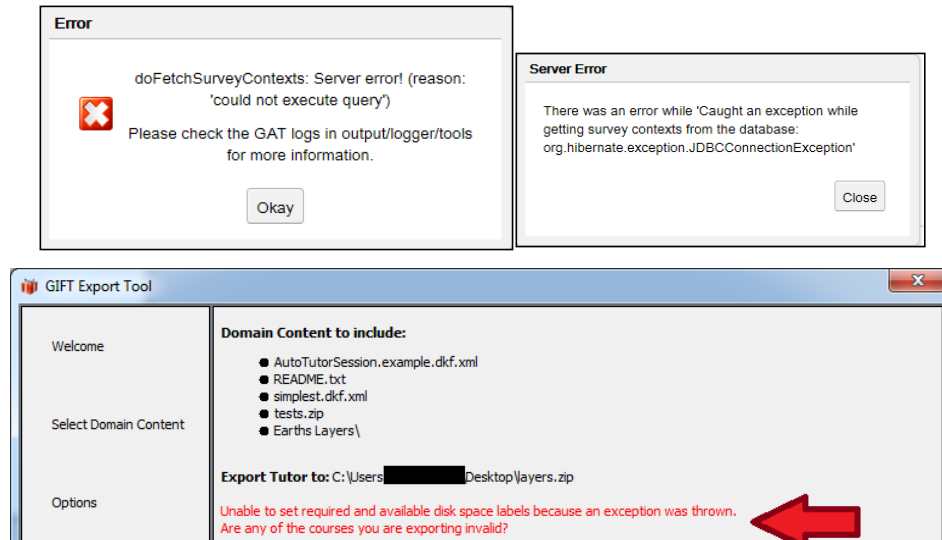
Fig. 12 Example error messages from course authoring (top left), survey authoring system (top right), and the course export tool (bottom)

Heuristic: help users recognize, diagnose, and recover from errors

Recommendation:

- Use plain language, precisely indicate the error, and suggest solutions.

2.2.12 Issue: Unfamiliar Phrasing of the GIFT Element DKF

Users may be unfamiliar with the term DKF as an acronym. Users may also be confused about how a Course and DKF interact with one another from a conceptual standpoint. This issue is closely related to the following issue.

2.2.13 Issue: Unclear Function Mapping between System Elements and ITS Components

Users familiar with the ITS model may be confused by the relationship of the GIFT DKF to ITS components/modules, specifically the domain module and pedagogical/tutoring module (Woolf 2009). The DKF help documentation states "the file is used by the Domain and Pedagogical modules". The name of this component, *domain knowledge file,* may be limited in the scope of the actual functionality of the component.

Heuristics: match between system and real world; recognition rather than recall

Recommendation:

- Consider renaming the DKF to a label that is more descriptive of its functionality and/or a label more familiar to users that may be familiar with ITS (as applicable).

2.2.14 Issue: No Specific Documentation for DKF Browser-Based Tool

Users may be confused regarding the use of the DKF web-based authoring tool. Although documentation exists on the DKF concept, no information was located that was specific to the use of the browser-based DKF Authoring Tool.

Heuristics: help and documentation; visibility of system status

Recommendation:

- Provide documentation specific to the browser-based DKF Authoring Tool.

2.2.15 Issue: Unintuitive Interface Layout for DKF Authoring Tool

Users may be confused and/or intimidated by the DKF Authoring Tool interface (Fig. 13). The complexity of the interface appears to be an order of magnitude greater than the Course Authoring Tool interface. However, none of the input fields appear to display help text on rollover. The State Transitions and Instructional Strategies UI frames appear blank when first visited. Open text fields (including some fields without validation) and system-labels appear throughout the UI.

Heuristics: match between system and the real world; help and documentation; recognition rather than recall.

Recommendation:

- Provide templates, examples, and on-demand help text for the DKF Authoring Tool.
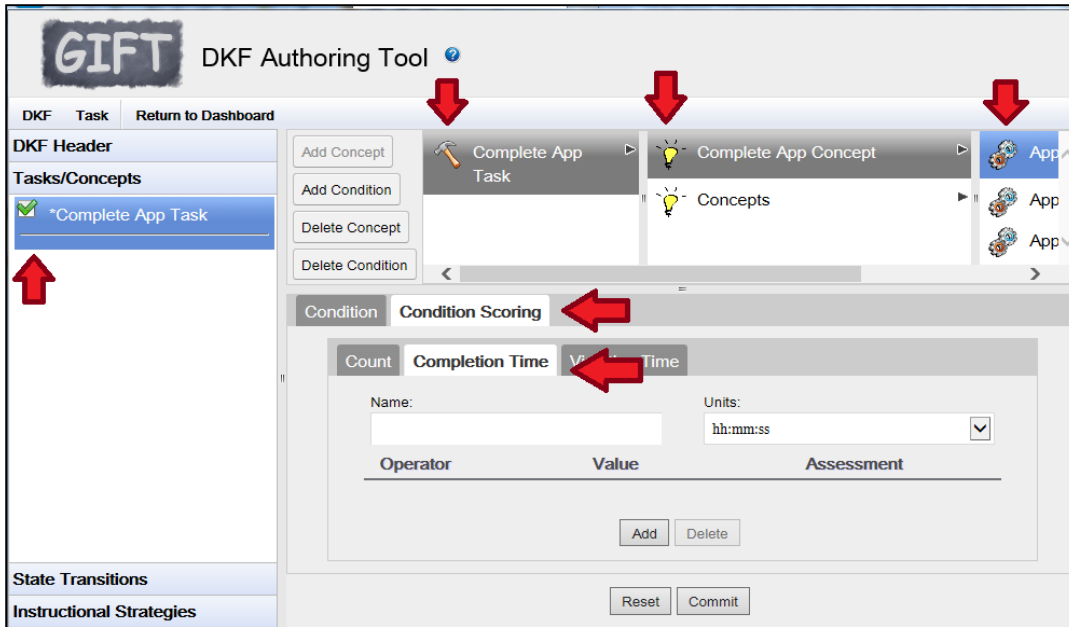
Fig. 13   DKF Authoring Tool consists of a series of nested UI elements

2.3 Task: Creating Surveys and Assessments with the GIFT Survey Authoring System (SAS)

2.3.1 Issue: Separate and Different Interfaces for Some Web-Authoring Tools

The GIFT Dashboard and the Survey Authoring System (SAS) appear as completely separate tools within the GIFT authoring experience. Further, the look and feel of those 2 interfaces are inconsistent with one another (Fig. 14). For instance, the SAS utilizes a row of tabs followed by a row of buttons, while the Dashboard uses a menu row followed by a row of tabs. These UI differences may lead to confusion when working between these tools. For novice GIFT authors, the learnability of the authoring tools and the overall GIFT framework may be negatively impacted as a result of the separation between the 2 tools.

Heuristic: consistency and standards

Recommendations:

- Integrate the SAS UI into the GIFT Dashboard (web).

- The SAS UI should be visually styled in a manner similar to the GIFT Dashboard.
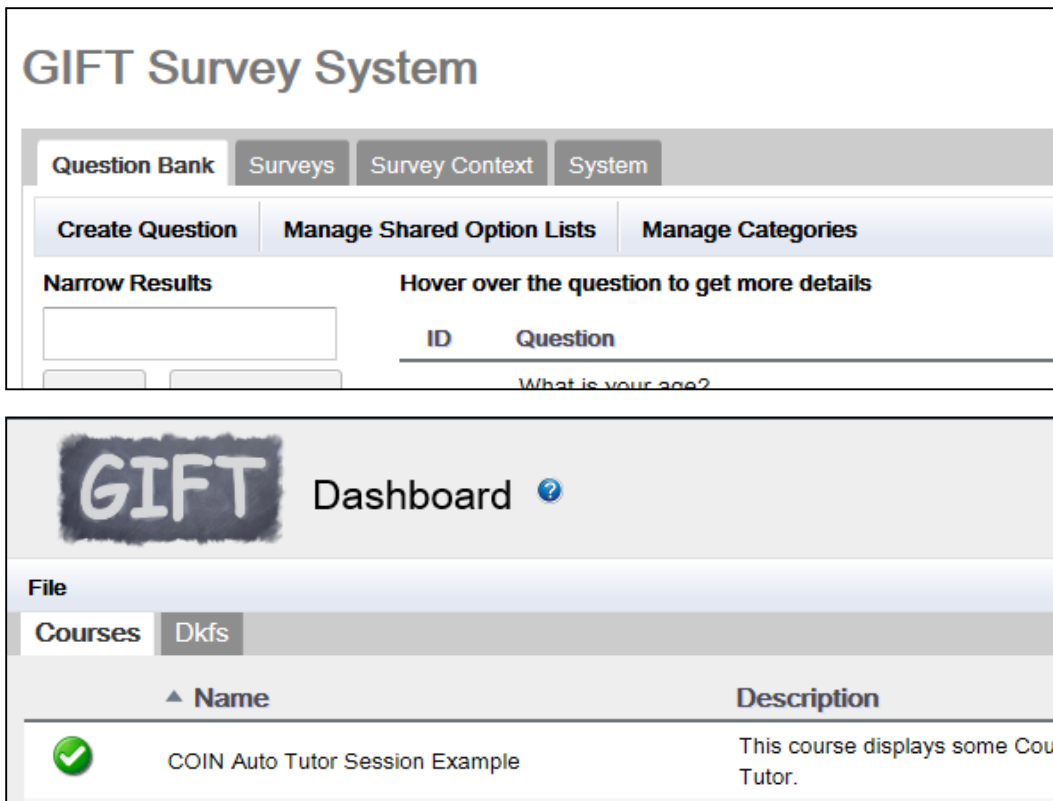
Fig. 14   Comparison of GIFT SAS and dashboard interfaces

2.3.2   Issue: Name Mismatch between Control Panel and Web UI

The survey authoring tool is referred to as the "Survey Authoring System (SAS)" within the GIFT Control Panel; the web interface that opens when the button is clicked is titled as the "GIFT Survey System" (Fig. 15).
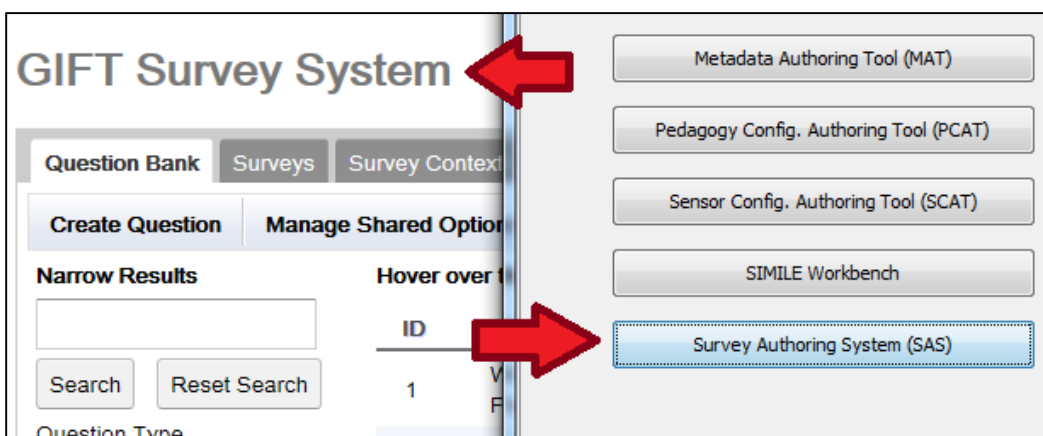
Fig. 15   Survey authoring web interface and control panel

Heuristic: consistency and standards

Recommendation:

- The naming of tools should be consistent throughout the authoring experience.

2.3.3 Issue: No Access to Help from within the SAS

Users may become frustrated when trying to locate support documentation from within the SAS. There does not seem to be to be any links or buttons within this interface that provide direct access to the documentation. By comparison, the GIFT Dashboard has a help icon to the right of the title at the top of each page (Fig. 14).

Heuristics: consistency and standards; help and documentation

Recommendation:

- Provide a link to documentation (opens in new tab) in the SAS.

2.3.4 Issue: Unfamiliar Terminology in the SAS

Users may be unfamiliar with the phrase "Survey Context" or its relationship to the Question Bank and Surveys in the SAS. Additionally, users may also be unfamiliar with "survey key" and "survey ID", which are attributes assigned to existing surveys within survey contexts. Information about these concepts is contained within the help documentation; however, it is not obvious to the user how these elements are used elsewhere in GIFT authoring.

Heuristics: match between system and real world; help and documentation

Recommendation:

- Provide on-demand guidance text within the survey context UI.

2.3.5 Issue: Confusing Workflow for Authoring within the SAS

Users may become confused with the correct procedure for creating a survey context to be used with a GIFT course. The SAS is organized into the components of questions, surveys, and contexts. Questions can be created within the Question Bank (orphans) or created within the Survey UI (as part of a Survey). By comparison, new surveys cannot be created directly within the Survey Context UI. Although the UI defaults to the Question Bank tab, the corresponding documentation instructs users to first "Ensure that all required surveys exist by identifying existing surveys that will be reused and by creating new (empty)

surveys if needed". However, the SAS does not permit the creation of empty surveys (Fig. 16). It is not clear if a top-down (e.g., survey context, surveys, then questions) or bottom-up (e.g., shared options lists, shared questions, surveys, keys, survey context) approach is appropriate for creating a series of surveys to be used within GIFT; the documentation seems to suggest starting somewhere in the middle with selection of existing survey content.

Heuristics: match between system and the real world; help and documentation; flexibility and efficiency of use

Recommendations:

- Allow the layout of the interface to suggest the correct workflow for the SAS.

- Provide templates that new users can "fill in" with the appropriate elements.

- Allow users to create skeleton outlines (with empty elements) where practical.
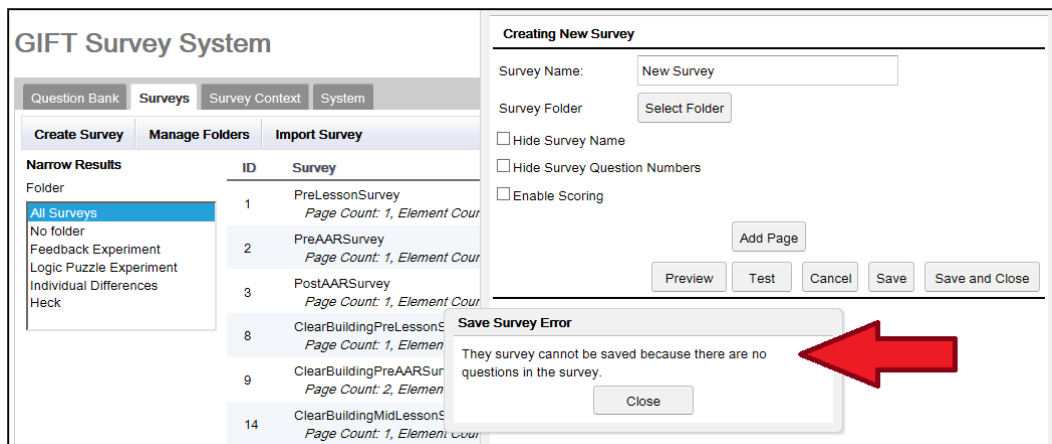
Fig. 16 Empty survey save error message

2.3.6 Issue: Unclear Dependency Status of Elements within the SAS

In the SAS, questions and surveys can be deleted, only if they are not linked to any surveys or survey contexts, respectively. When attempting to delete an element with a dependency, the system will inform the user that the element cannot be deleted. However, the system does not indicate which other elements in the system are dependent upon the target element.

Heuristics: visibility of system status; help users recognize, diagnose, and recover from errors

Recommendations:

- Add a dependency column to the SAS UI; clicking on this column will display a list of parent elements that are dependent upon the target element.

- Add dependency information to message boxes (e.g., cannot delete this question, the following surveys are dependent upon this element…).

2.3.7 Issue: Unclear Status of Survey Context, Data Management

Users may become confused about how to view survey responses for a given survey context. The SAS currently allows users to Erase Survey Responses within the Survey Context tab by clicking on the Edit button for a target context (Fig. 17). However, there does not appear to be a mechanism for reviewing survey responses within the SAS. The UI does not indicate if any survey response data currently exists for a given survey context. Without prior knowledge of the current state of the survey context, users may not be aware of what, if anything, is being erased using this function.
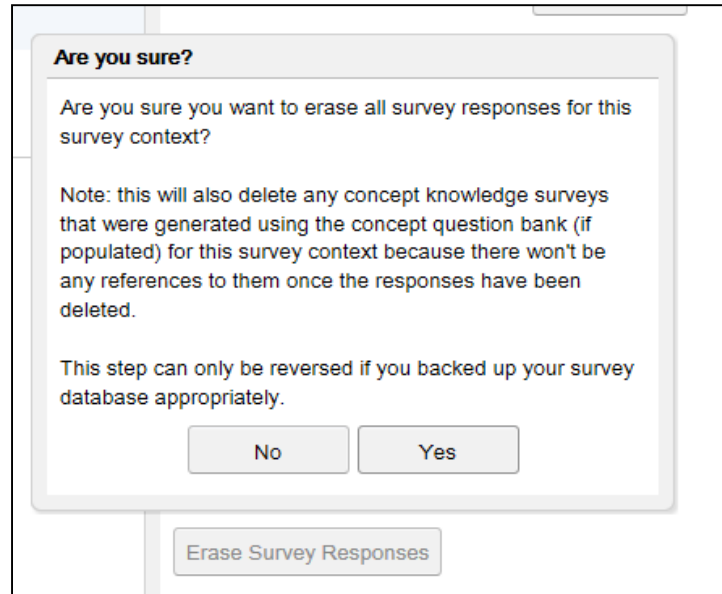
Fig. 17 Confirmation dialog to erase survey response

Heuristics: visibility of system status; recognition rather than recall

Recommendations:

- Add a column to the Survey Context UI to indicate how many responses have been recorded for each survey context.

- Allow users to review survey responses from within the SAS UI.

- Remove the ability for users to erase survey responses from within the SAS UI; create a separate interface/tab for data management.

- If survey contexts can be assigned to different courses, the user should be able to erase a subset of survey responses by course or session.

2.4 Task: Using Other Tools in the GIFT Authoring Toolset

2.4.1 Issue: AutoTutor Script Authoring Tool Requires Different Credentials

Users may become frustrated when attempting to access the AutoTutor Script Authoring Tool (ASAT). Access to the tool requires users to have a set of credentials that is different from those established on the GIFT Tutoring website (www.gifttutoring.org). Specifically, Google account credentials are required to access the tool. The pop-up message on the first page of the ASAT does not inform the user that a third-party login is required (Fig. 18). The use of third-party

credentials may disrupt the GIFT experience, and may also raise privacy concerns for users logging in with their personal Google accounts.

Heuristic: consistency and standards

Recommendation:

- Allow users to log into the ASAT with GIFT Tutoring credentials; or remove the login requirement.
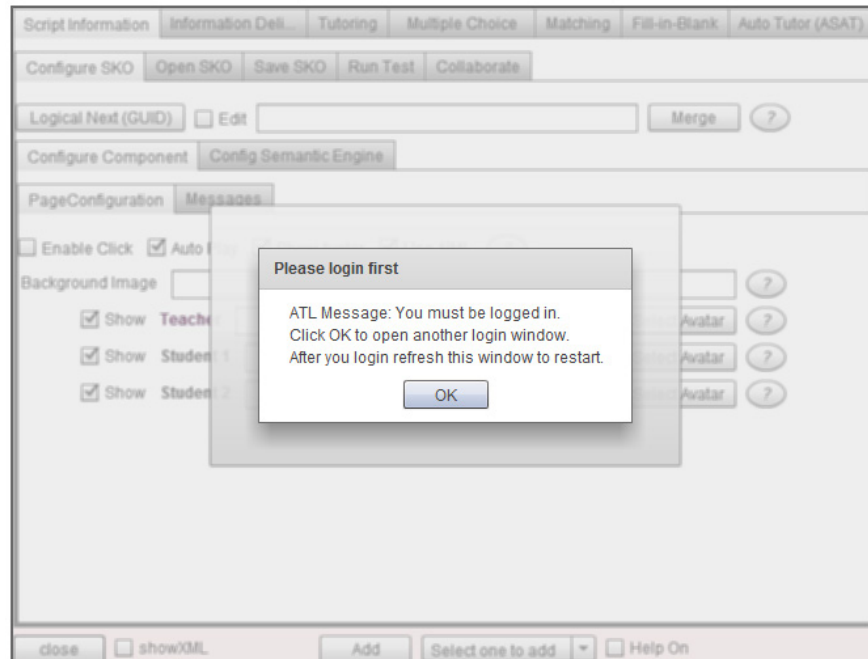
Fig. 18 Initial screen of AutoTutor script authoring tool

2.4.2 Issue: eMAP Name Mismatch between Documentation and Control Panel

The Engine for Management of Adaptive Pedagogy (eMAP) does not have a specific button callout or section within the GIFT Control Panel. The Pedagogy Config. Authoring Tool appears to open the primary eMAP authoring interface; however, authors might need to at least access the Metadata Authoring Tool to generate a course that leverages the eMAP technology. The Learner Config. Authoring Tool, and/or the Sensor Config. Authoring Tool may also need to be used under specific circumstances. Users may be confused by the correct tool or set of tools required to leverage the eMAP.

Heuristics: consistency and standards; help and documentation

Recommendations: (for Control Panel)

- Consider renaming the Pedagogy Config. Authoring Tool to eMAP Authoring Tool.

- Consider visually grouping the buttons for tools that are critical to using the Pedagogy Config. Authoring Tool for faster user recognition (Fig. 19).
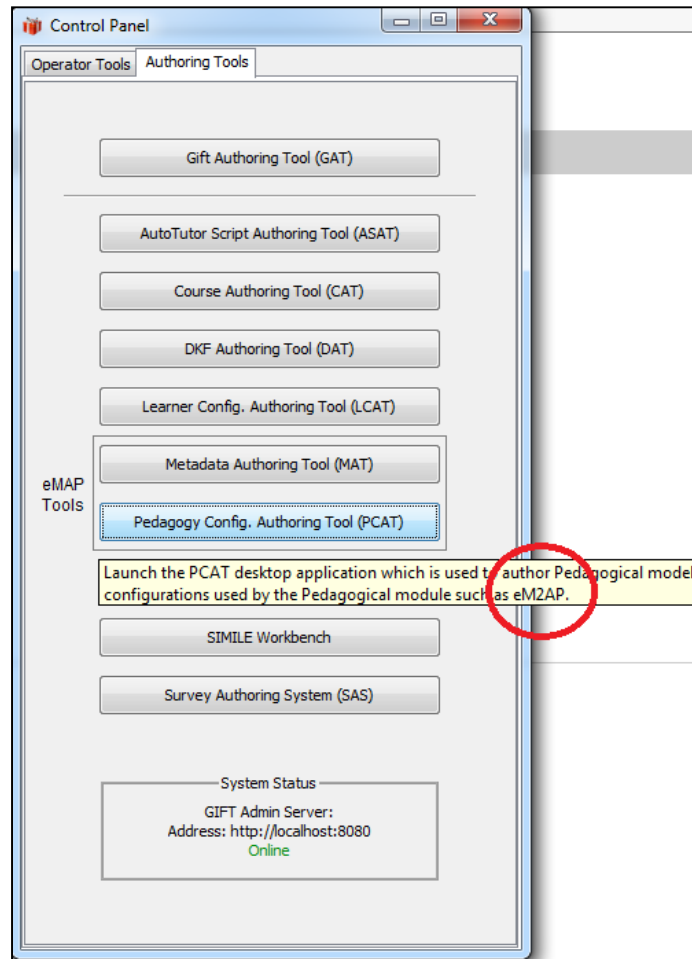
Fig. 19   Example grouping of eMAP tools (typo circled)

2.4.3   Issue: Name Mismatch between SIMILE Workbench and Control Panel

The SIMILE workbench that launches from the Control Panel is titled "Simile Workbench". The tool is referred to as the SIMILE Workbench (caps) elsewhere in the GIFT software (Fig. 20).

Heuristic: consistency and standards

Recommendation:

- All references to that tool appear as SIMILE Workbench.
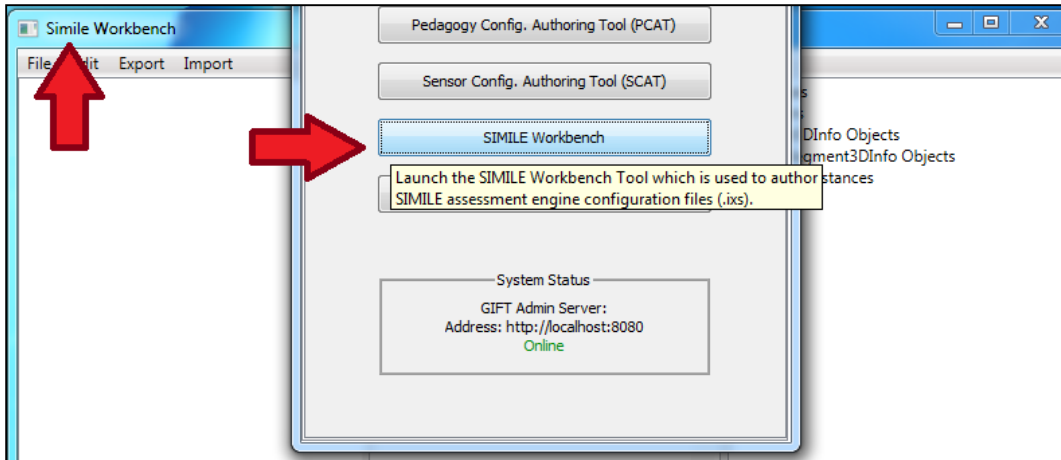
Fig. 20   Layout and labeling of SIMILE workbench

2.4.4 Documentation Not Found for SIMILE Workbench

The SIMILE Workbench does not contain access to help from the tool's UI (Fig. 20). The GIFT support documentation does not appear to contain any documentation on the configuration and use of the SIMILE Workbench. The purpose of the SIMILE Workbench, as well as how the tool fits into the overall GIFT authoring workflow, may not be clear to the user.

Heuristics: visibility of system status; help and documentation

Recommendations:

- Provide documentation on the use of the SMILE Workbench.

- Provide quick access to SIMILE Workbench documentation from the tool's UI.

3.   Discussion of Findings

The findings of this evaluation revealed a number of key usability concerns that warrant further investigation. The following sections are organized by usability principles.

3.1 Mapping between Conceptual Model and GIFT Implementation

The results of the evaluation suggest that users may find it difficult to translate their desired authoring task goals into actions and intentions using the GIFT authoring tools. Creating an adaptive tutor is an inherently complex task; it is likewise difficult to infer the conceptual model for GIFT from the system

representation of the visible GIFT authoring tools. There appears to be a considerable gap between the courses that a user might want to build with GIFT and how courses are actually constructed using the GIFT authoring tools. Norman refers to this gap as the "gulf of execution" (Norman 1986).

Further complicating matters, there are arguably many mental models that a potential user of the GIFT Authoring tools may apply to the construction of an adaptive course. These mental models, in theory, help to shape the way in which an author understands and approaches the GIFT architecture. For instance, an instructor may be familiar with the systems model approach (Fig. 21) for designing instruction (Dick et al. 2005), but not the adaptive tutoring model (Fig. 22).

Fig. 21   Systems approach model for designing instruction

Fig. 22   Adaptive tutoring learning effect model (Sottilare and LaViola 2015)

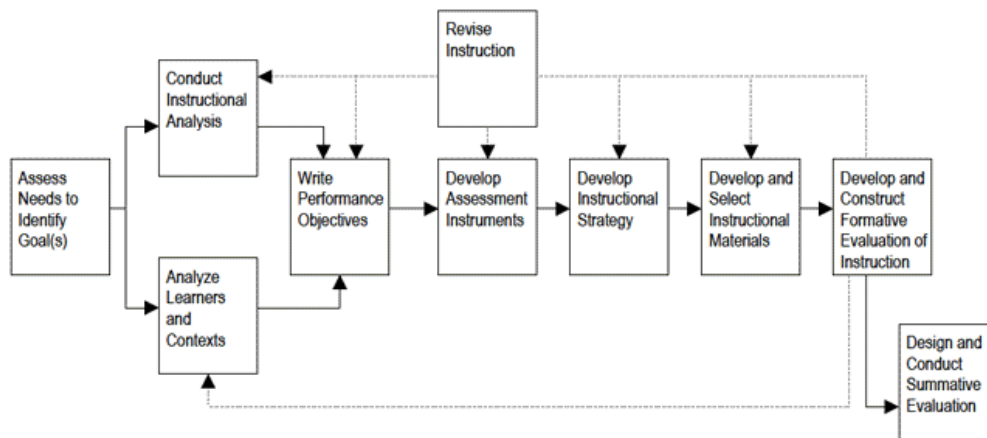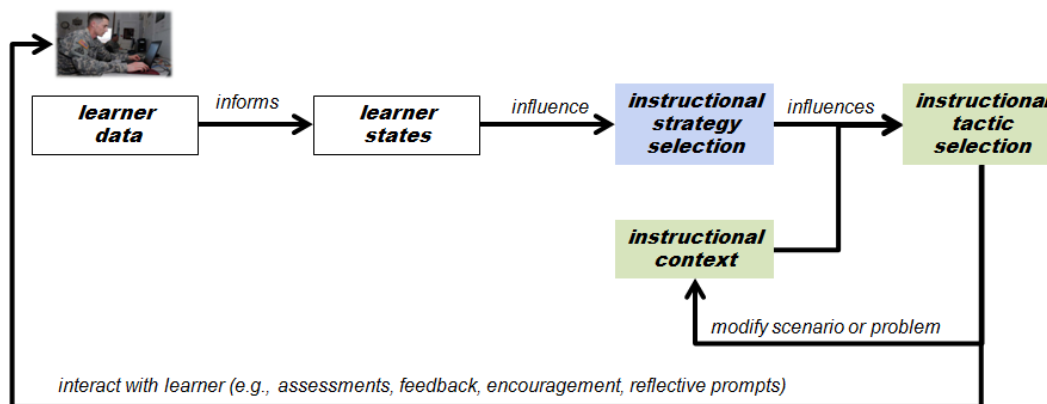However, those models are not representative of specifically how course transitions, survey contexts, sensors, metadata, and DKFs interoperate within GIFT. Importantly, those potential mental models may reflect the expectations a user might have about the functionality of the GIFT authoring tools and assumptions about how those authoring tools should be used. For instance, a subject matter expert may not be familiar with either model but may expect GIFT authoring tools to function similarly to familiar content-creation applications such as Microsoft PowerPoint.

Two example approaches are presented for closing the gap between user intent and system functionality. The first approach involves configuring the UI to better convey the relationships between GIFT components and tools, through either a more robust course layout (e.g., flowchart or process diagram) or a wizard-style interface. A GIFT system diagram might also benefit the user's understanding of the system, one that demonstrates the exchange of information between the different GIFT-specific modules. Alternatively, the second approach involves redesigning the UI to align more closely with users' potential mental models of tutor authoring. In this approach, existing tools are replaced with new tools informed by user-centered design. (for example, the Metadata Authoring Tool becomes absorbed into a more user-friendly Content Repository Tool).

3.2 Layout and Appearance of Authoring Tools

The GIFT authoring experience consists of a number of physically separate interfaces and tools. Because of this, it may be very difficult for a user to ascertain the relationships between the elements created by each tool, or the order in which the tools should be used in an average GIFT authoring workflow. Some GIFT authoring tools begin with a blank slate, without any templates other than the example courses that are included in GIFT. This may be confusing or even intimidating to users who wish to learn how to link a survey to a course, or leverage unique GIFT capabilities such as the eMAP or a DKF.

Further, the user will encounter functionally similar but visually different layouts when working between the GIFT Dashboard (Course Authoring Tool and DKF Authoring Tool) and the Survey Authoring System (SAS), respectively. The current set of *all* authoring tools consist of 3 visually different browser-based interfaces (e.g., AutoTutor Script Authoring Tool), XML editors, and a custom interface for the SIMILE Workbench. Those distinct interfaces may detract from the overall perceived cohesion of GIFT and may unnecessarily increase the effort required to author a course due to the time spent switching between interfaces.

It is recommended that the entire GIFT authoring experience appear within a single application. For instance, different tools and "views" of the tutor could be represented by different tabs on the UI, similar to the ribbon UI of Microsoft Office applications. Changes made to the system in one tab should update and populate fields on other tabs/views as appropriate. The extensible nature of GIFT should allow new tabs to be added to the UI as new technologies mature and become available (e.g., downloadable plug-ins, add-ins, or extensions).

3.3 Internal Consistency of Names and Labels

The evaluation revealed that many names and labels that were not internally consistent throughout the GIFT authoring tools. For example, the Control Panel refers to the Gift [sic] Authoring Tool (GAT), where GIFT is represented in initial caps. Clicking this button launches a window titled, "GIFT Dashboard". Similarly, clicking the "Survey Authoring System (SAS)" button on the Control Panel launches a window titled, "GIFT Survey Tool". Further, many of the tools referenced on the Control Panel carry the suffix "Authoring Tool" with a corresponding acronym. Many of these acronyms do not seem to appear elsewhere within GIFT, making the tool names unnecessarily long with acronyms that users may not be able to remember.

3.4 Match between System Language and Real-World Language

The evaluation identified a number of phrases and concepts with which users may be unfamiliar, including: DKF, eMAP, SIMILE, Transitions, Interops, and Survey Context. Further, there are instances in GIFT where long path names or strings of system information are used in the configuration of a course (e.g., names of Interop Implementations found within the Training App course transition). The use of technical and system-level language increases skill threshold required for novice users to learn how to use the GIFT authoring tools, as well as increases the potential for errors while authoring. It may benefit the user to update these phrases and system strings with user-friendly labels that more fully and accurately convey their meaning or purpose.

3.5 Error Prevention and Preview

The results of the evaluation suggest that the potential for authoring error is high, while error tolerance in the GIFT authoring tools is low. The DKF and Course Authoring Tools appear to enable and make available all of the features and options in course authoring, even if the user has not installed certain components or is not using certain training applications. Many fields are not evaluated for

erroneous input. Certain input fields will generate error messages, though the content of these messages sometimes contain system-level information or will refer users to the external file system to review log files.

Further, aside from the SAS, the GIFT authoring tools do not currently provide for an easy way to preview a course as it is being created. The lack of that functionality runs counter to the stated goal of rapid prototyping of adaptive tutoring. Even if the inherent complexity of GIFT precludes the authoring tools from being able to reliably error check user input or enable/disable extraneous sections, allowing users quickly preview their additions and edits may help authors to quickly self-correct errors without exiting authoring experience.

3.6 File System, Organization, and Reuse

The user is largely responsible for managing their content and course files when interacting with GIFT. The documentation for GIFT recommends that courses and materials should be organized into subfolders, but the system does not allow the user to create subfolders when saving. Additionally, GIFT currently lacks a complete experience for managing and tagging (i.e., metadata) content. With respect to local content, the user must transfer media files into the course domain file location to include that media material when courses are exported from GIFT. These issues impede authors' ability to reuse content and do not support users in organizing their knowledge. The expected future integration with a formal content management system should help to alleviate many of these issues, in addition to standardizing the integration of external media and applications.

3.7 Uneven Help and Documentation Experience

The evaluation revealed inconsistent access to help and documentation throughout the various GIFT authoring tool interfaces. For example, many of the fields in the Course Authoring Tool are annotated with mouse-over text, although no assistive text appears within the DKF authoring tool. The GIFT Dashboard provides a link to the documentation, while there is no link to the documentation within the SAS. Also, the available documentation does not fully cover all of the authoring tools that are currently present. For example, documentation is available with regard to DKF authoring in XML, but no specific documentation is available for the browser-based version of the DKF authoring tool. A complete set of up-to-date documentation and robust on-demand help within the authoring interface(s) will help to decrease the effort and skill threshold required for authoring adaptive tutoring.

4. Conclusion

This heuristic evaluation of the GIFT Authoring Tools (version 2014-3X) revealed a number of usability issues that are driving additional development and future testing with GIFT users. Notable usability concerns include the multiple interfaces that make up the current authoring tool suite and potential user confusion of the overall integration of the tools and authoring workflow in GIFT. Other concerns include the technical and system nature of the language used throughout the authoring tools and the lack of internal consistency with respect to terminology and labeling between the multiple interfaces. GIFT documentation should also be tailored for specific implementations of GIFT (e.g., browser-based tools). Finally, this evaluation identified the need for a more robust help and documentation experience that continues to be updated as new versions of GIFT are developed and released.

A major design issue related to the user's interaction with the GIFT authoring tools is the use of separate tools and interfaces for each of the major functions in the authoring process. This was a reasonable outcome of a research project to investigate each of the elements required for effective authoring of ITSs, but is a drawback to usability. GIFT 2014-3X contains 9 separate XML-based authoring tools for the functions within the authoring process:

- **AutoTutor Script Authoring Tool (ASAT):** used to generate dialogue-based tutoring sessions to support interaction between a virtual human tutor and the user

- **Course Authoring Tool (CAT):** used to sequence and launch transitions (e.g., guidance, surveys, material, training applications, after-action reviews, or Merrill's branch points) for a course, module, or lesson

- **Domain Knowledge File (DKF) Authoring Tool (DAT)**: used to identify behaviors and measures and their relationship to domain learning objectives or concepts within GIFT to support assessment of the learner

- **Learner Configuration Authoring Tool (LCAT)**: used to author configuration files used by the learner module to format the learner data sensor pipeline

- **Metadata Authoring Tool (MAT)**: used to author files that contain attributes describing domain content files; these attributes are used by GIFT to select appropriate content for presentation to the user based on their states/traits

- **Pedagogy Configuration Authoring Tool (PCAT)**: used to author configurations used by the Pedagogical Module to determine appropriate instructional strategies based on user states; the default configuration for the Pedagogical Module configuration is the engine for Managing Adaptive Pedagogy (eMAP)

- **Sensor Configuration Authoring Tool (SCAT):** used to author sensor configuration files used by the GIFT sensor module to manage the flow and format of sensor data and associated filters and writers

- **Student Information Models for Immersive Learning Environments (SIMILE) Workbench Tool:** used to map configurations detailing the relationship between behaviors in a simulation (e.g., serious game) and their relationship to progress toward learning objectives

- **Survey Authoring System (SAS):** used to author surveys and assessments that can be presented to the user during course execution

Each of these tools provides an essential function within the authoring process for adaptive training, but a comprehensive tool is needed to guide the user through the end-to-end authoring process. The user should not be expected to recognize and understand the functionality of 9 tools. Instead, we are working toward the goal of a single tool, the GIFT Authoring Tool (GAT), which relies on artificial intelligence to provide the user with guidance and feedback on the development of their tutor through a simple front-end or dashboard that uses the function and power of the current tools to execute functions in the background. A guided interview process is envisioned to elicit the information needed to construct the tutor and a dashboard to convey progress to the user. The findings and recommendations within this report are intended to support the future development of GIFT authoring capabilities and inform formal usability testing of the authoring tools with representative user groups. Since GIFT is part of a spiral development process (e.g., model-test-model), it is anticipated that usability evaluations will be needed periodically as the technical goals are achieved as delineated in the research outlines that follow.

- ARL Special Report (ARL-SR-0325): Domain Modeling for Adaptive Training and Education in Support of the US Army Learning Model—Research Outline (Sottilare et al. 2015)

- ARL Special Report (ARL-SR-0333): Effectiveness Evaluation Tools and Methods for Adaptive Training and Education in Support of the US Army Learning Model: Research Outline (Johnston et al. 2015)

- ARL Special Report (ARL-SR-0336): Individual Learner and Team Modeling for Adaptive Training and Education in Support of the US Army Learning Model: Research Outline (Goodwin et al. 2015)

- ARL Special Report (ARL-SR-0339): Authoring Tools and Methods for Adaptive Training and Education in Support of the US Army Learning Model: Research Outline (Ososky et al. 2015)

- ARL Special Report (ARL-SR-0345): Instructional Management for Adaptive Training and Education in Support of the US Army Learning Model–Research Outline (Goldberg et al. 2015)

5.   References

Dick W, Carey L, Carey JO. The systematic design of instruction. 6th ed. Boston (MA): Pearson Allyn and Bacon; 2005.

GIFTTutoring.   GIFT   overview.   [accessed   2015   Feb   27]. https://www.gifttutoring.org/projects/gift/wiki/Overview.

Goldberg B, Sinatra A, Sottilare R, Moss J, Graesser A. Instructional management for adaptive training and education in support of the US Army learning model–research outline. Adelphi (MD): Army Research Laboratory (US); 2015 Nov. Report No.: ARL-SR-0345.

Goodwin G, Johnston J, Sottilare, R, Brawner K, Sinatra, A, Graesser, A. Individual learner and team modeling for adaptive training and education in support of the US Army learning model: research outline. Adelphi (MD): Army Research Laboratory (US); 2015 Sep. Report No.: ARL-SR-0336.

Johnston J, Goodwin G, Moss J, Sottilare R, Ososky S, Cruz D, Graesser A. Effectiveness evaluation tools and methods for adaptive training and education in support of the US Army learning model: research outline. Adelphi (MD): Army Research Laboratory (US); 2015 Sep. Report No.: ARL-SR-0333.

Nielsen J. Heuristic evaluation. In: Nielsen J, Mack RL, editors. Usability inspection methods. New York (NY): John Wiley and Sons; 1994.

Nielsen J. 10 usability heuristics for user interface design. Freemont (CA): Nielsen Norman Group; 1995 Jan 1. [accessed 2015 Feb 27]. http://www.nngroup.com/articles/ten-usability-heuristics/.

Norman DA. Cognitive engineering. In: Norman DA, Draper SW, editors. User centered system design: new perspectives on human-computer interaction. Hillsdale (NJ): Lawrence Erlbaum Associates; 1986. p. 31–61.

Ososky S, Sottilare R, Brawner K, Long R, Graesser A. Authoring tools and methods for adaptive training and education in support of the US Army learning model: research outline. Adelphi (MD): Army Research Laboratory (US); 2015 Oct. Report No.: ARL-SR-0339.

Sottilare R, LaViola J. Extending intelligent tutoring beyond the desktop to the psychomotor domain: a survey of smart glass technologies. In: Proceedings of the Interservice/Industry Training Simulation and Education Conference; 2015 Nov 30–Dec 4; Orlando, FL.

Sottilare RA, Brawner KW, Goldberg BS, Holden HK. The Generalized Intelligent Framework for Tutoring (GIFT). Orlando (FL); US Army Research Laboratory–Human Research and Engineering Directorate (HRED); 2012.

Sottilare RA, Graesser AC, Hu X, Holden H. Design recommendations for intelligent tutoring systems: volume 1, learner modeling. Orlando (FL): US Army Research Laboratory; 2013. Available at: https://gifttutoring.org/documents/42.

Sottilare R, Sinatra A, Boyce M, Graesser A. Domain modeling for adaptive training and education in support of the US Army learning model—research outline. Adelphi (MD): Army Research Laboratory (US); 2015 Jun. Report No.: ARL-SR-0325.

Woolf, BP. Building interactive intelligent tutors: student-centered strategies for revolutionizing e-learning. Burlington (MA): Morgan Kaufmann Publishers; 2009.

List of Symbols, Abbreviations, and Acronyms

ARL	US Army Research Laboratory
ASAT	AutoTutor Script Authoring Tool
CBTS	computer-based tutoring systems
DAT	DKF Authoring Tool
DKF	domain knowledge file
eMAP	Engine for Management of Adaptive Pedagogy
GAT	GIFT Authoring Tool
GIFT	Generalized Intelligent Framework for Tutoring
ITS	intelligent tutoring system
SAS	Survey Authoring System
TUI	tutor-user interface
UI	user interface
URI	Uniform Resource Identifier
XML	eXtensible Markup Language

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDF)	DIRECTOR US ARMY RESEARCH LAB RDRL CIO LL IMAL HRA MAIL & RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
2 (PDF)	DIR USARL RDRL HR R SOTTILARE RDRL HRT S OSOSKY